Available online at www.sciencedirect.com
ScienceDirect
Journal of Complexity 22 (2006) 818–826
Journal of COMPLEXITY
www.elsevier.com/locate/jco

A hierarchy of Turing degrees of divergence bounded computable real numbers☆

Robert Rettinger[a], Xizhong Zheng[b, c, *]

[a] Theoretische Informatik II, FernUniversität Hagen, 58084 Hagen, Germany
[b] Department of Computer Science, Jiangsu University, Zhenjiang 212013, China
[c] Theoretische Informatik, BTU Cottbus, 03044 Cottbus, Germany

Received 12 December 2005; accepted 25 April 2006
Available online 7 July 2006

Abstract

A real number x is f-bounded computable (f-bc, for short) for a function f if there is a computable sequence (x_s) of rational numbers which converges to x f-bounded effectively in the sense that, for any natural number n, the sequence (x_s) has at most $f(n)$ non-overlapping jumps of size larger than 2^{-n}. f-bc reals are called divergence bounded computable if f is computable. In this paper we give a hierarchy theorem for Turing degrees of different classes of f-bc reals. More precisely, we will show that, for any computable functions f and g, if there exists a constant $\gamma > 1$ such that, for any constant c, $f(n\gamma) + n + c \leqslant g(n)$ holds for almost all n, then the classes of Turing degrees given by f-bc and g-bc reals are different. As a corollary this implies immediately the result of [R. Rettinger, X. Zheng, On the Turing degrees of the divergence bounded computable reals, in: CiE 2005, June 8–15, Amsterdam, The Netherlands, Lecture Notes in Computer Science, vol. 3526, 2005, Springer, Berlin, pp. 418–428.] that the classes of Turing degrees of d-c.e. reals and divergence bounded computable reals are different.

Keywords: Turing degree of reals; Divergence bounded computable real; Hierarchy

1. Introduction

In order to investigate computability of real numbers we usually look at convergent computable sequences (x_s) of rational numbers. Depending on how fast a sequence converges, we distinguish several computability levels of its limit. For example, in the optimal case, if (x_s) converges

☆ This work is supported jointly by DFG (446 CHV 113/240/0-1) and NSFC (10420130638).

* Corresponding author. Theoretische Informatik, BTU Cottbus, 03044 Cottbus, Germany.
E-mail address: zheng@informatik.tu-cottbus.de (X. Zheng).

0885-064X/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jco.2006.04.009

effectively in the sense that $|x_s - x_{s+1}| \leqslant 2^{-s}$ for all s, then its limit x is called *computable* [8,6]. If the sequence converges *weakly effectively* in the sense that the sum $\sum_{s\in\mathbb{N}} |x_s - x_{s+1}|$ is finite, then its limit is *weakly computable* [1] which is a weaker computability notion than that of computable reals. Moreover, if we do not ask for additional conditions besides computability and convergence of the sequences, then the limits are called *computably approximable* (c.a.) or Δ^0_2 [1,9]. In general, we measure the convergence speed of a sequence by counting the number of jumps of certain sizes. Two different types of jumps can be considered here, depending on how we classify the size of a jump, namely, the jumps (x_i, x_j) with $2^{-n} \leqslant |x_i - x_j| < 2^{-n+1}$ or with $|x_i - x_j| \geqslant 2^{-n}$ for some $n \in \mathbb{N}$.

According to the first type of jumps, the authors introduced in [12] the Cauchy computability of reals. A real x is called *f-Cauchy computable* if there exists a computable sequence (x_s) of rational numbers which converges to x *f-effectively*, which means that there are at most $f(n)$ non-overlapping pairs of indices (i, j) such that $2^{-n} \leqslant |x_i - x_j| < 2^{-n+1}$ for all $n \in \mathbb{N}$. (If f is a computable function, then f-Cauchy computable real numbers are also called *ω-Cauchy computable*.) Concerning the f-Cauchy computability there is a hierarchy theorem similar to Ershov's Hierarchy theorem [4] of subsets of natural numbers. That is, if f and g are different in infinitely many places, then the classes of f- and g-Cauchy computable reals are different.

Using the second type of jumps, the f-bounded computability is introduced in [10]. More precisely, we call an index pair (i, j) a 2^{-n}*-jump* if $|x_i - x_j| \geqslant 2^{-n}$. A sequence (x_s) converges *f-bounded effectively* if it has at most $f(n)$ non-overlapping 2^{-n}-jumps for all $n \in \mathbb{N}$. A real number x is *f-bounded computable* (f-bc, for short) if there is a computable sequence (x_s) of rational numbers which converges to x f-bounded effectively. Naturally, we need only to consider monotone functions f here. If a real number is f-bc for a computable function f, then it is also called *divergence bounded computable* (dbc, for short). Obviously, the divergence bounded computable reals, are exactly the ω-Cauchy computable reals while f-bounded and f-Cauchy computability are different notions in general. For f-bc reals, we have another kind of hierarchy theorem (see [10]) that, if the computable functions f and g have an unbounded distance, i.e., $(\forall c \in \mathbb{N})(\exists n)(|f(n) - g(n)| \geqslant c)$, then the classes of f- and g-bounded computable reals are different. A very nice property of f-bc reals has been shown in [10]: if C is a class of functions such that for any $f, g \in C$ and $c \in \mathbb{N}$ there is an $h \in C$ so that $h(n) \geqslant f(n + c) + g(n + c)$ for all n, then the class of all C-bc (f-bc for an $f \in C$) reals is a field.

Intuitively, if $f(n) \leqslant g(n)$ for all n, then an f-bc real should not be more complicated than a g-bc real even from the (classical) computability point of view. To give some evidence for this we consider Turing degrees as a measure for complexity of real numbers, where the Turing degree of a real number is purely based on its binary expansion. Without loss of generality we only consider reals in the unit interval [0; 1]. For any real $x \in [0; 1]$ there exists a set $A \subseteq \mathbb{N}$ such that $x = x_A := \sum_{i\in A} 2^{-(i+1)}$. The set A is called the *binary expansion* of the real x_A because the binary expansion of x_A is $0.A$ if A is identified with its characteristic sequences. A real x_A is called *Turing reducible* to another real x_B (denoted by $x_A \leqslant_T x_B$) if the set A is Turing reducible to the set B, i.e., $A \leqslant_T B$ and two reals x, y are *Turing equivalent* (denoted by $x \equiv_T y$) if $x \leqslant_T y \& y \leqslant_T x$. Finally, the equivalence class $\deg_T(x) := \{y : x \equiv_T y\}$ is called the *Turing degree* of x. For convenience, we do not distinguish between Turing degrees of real numbers x_A and Turing degrees of the sets A. Thus, a Turing degree of reals can be called *c.e.* or *ω-c.e.* if it contains a c.e. or ω-c.e. set. Here, a set A is called ω-c.e. if there is a computable sequence (A_s) of finite sets which converges to A such that $|\{s : A_s(n) \neq A_{s+1}(n)\}| \leqslant h(n)$ for all $n \in \mathbb{N}$ and some computable function h, i.e., A can be approximated in such a way that the membership of a natural number n can be changed at most $h(n)$ times.

The computable enumerability was also introduced for real numbers. A real number x is called *computably enumerable* (c.e.) or *left computable* if there exists an increasing computable sequence of rational numbers which converges to x. Obviously, any real with a c.e. binary expansion (so called *strongly c.e.* real by Downey [2]) is a c.e. real but not every c.e. real has a c.e. binary expansion (cf. [7]). However, since the left Dedekind cut of a c.e. real is a c.e. set and it is Turing equivalent to the binary expansion of this real, the Turing degree of any c.e. real contains at least a c.e. set. That is, the class of c.e. Turing degrees (the degrees containing a c.e. set) is just the class of Turing degrees which contain at least one c.e. real. However, if we consider the d-c.e. degrees, the situation is different. A real number is called *d-c.e.* (difference of c.e.) if it is the difference of two c.e. reals. D-c.e. reals are exactly the weakly computable reals [1]. In [11] Zheng shows that there is a Turing degree containing a d-c.e. real which does not contain any ω-c.e. sets. Moreover, Downey et al. [3] show that every ω-c.e. Turing degree contains a d-c.e. real but not every Δ^0_2-degree contains a d-c.e. real. Recently, the authors [5] show that the classes of Turing degrees of d-c.e. reals, dbc reals and c.a. reals, respectively, are all different.

In this paper we investigate the hierarchy of Turing degrees of real numbers more systematically. Our aim is to find sufficient conditions on f and g to separate the classes of Turing degrees of f-bc and g-bc reals. As mentioned before, if two computable functions f and g satisfy the condition $(\forall c)(\exists^\infty n)(f(n) + c \leqslant g(n))$, then there is a g-bc real which is not f-bc. For the Turing degrees we will show that a strengthening suffices to separate even the classes of Turing degrees. Namely, if there is a constant $\gamma > 1$ such that for any constant c, the inequality $f(n\gamma) + n + c \leqslant g(n)$ hold for almost all n, then there is a g-bc real which is not Turing equivalent to any f-bc real.

Notice that any d-c.e. real is $\lambda n.2^n$-bounded computable. Let $f(n) := 2^n$ and $g(n) := 2^{2n}$ for all n. Then f and g satisfy the above condition and hence there is a g-bc real which is not Turing equivalent to any d-c.e. reals. This implies immediately the result of [3,5] that there is a dbc real which is not Turing equivalent to any d-c.e. reals.

2. Preliminaries

In this section we explain some notations and prove a technical lemma which we will use in the proof of our main theorem.

Let $\mathbb{N}$, $\mathbb{D}$, $\mathbb{Q}$ and $\mathbb{R}$ be the classes of natural, dyadic rational, rational and real numbers, respectively. Let $\langle \cdots \rangle$ be a computable n-pairing function and let $\pi_1, \dots, \pi_n$ be its inverse functions. Let $\Sigma := \{0, 1\}$ be a binary alphabet. The classes of finite strings and infinite sequences of Σ are denoted by Σ^* and Σ^ω, respectively. For convenience, we identify a real $x \in [0, 1]$ with its binary characteristic sequence $x \in \Sigma^\omega$ and identify a dyadic rational number $r \in [0, 1]$ with its binary characteristic string $r \in \Sigma^*$. For any binary string w, an open interval of w is defined by $I(w) := w\Sigma^\omega \setminus \{w1^\omega, w0^\omega\}$. For $I := I(w)$ and $n \in \mathbb{N}$, we define two subintervals $L_n(I) := I(w00^n1)$ and $R_n(I) := I(w10^n1)$. Especially, we denote $L(I) := L_1(I)$ and $R(I) := R_1(I)$. Thus, $L(I)$ and $R(I)$ are the second and sixth subintervals, respectively, if the interval I is divided equidistantly into eight subintervals. Let I, J be any intervals, their minimal and maximal distances are denoted by $d(I, J) := \min\{|x - y| : x \in I \;\&\; y \in J\}$ and $D(I, J) := \max\{|x - y| : x \in I \;\&\; y \in J\}$, respectively. Given two intervals $I_1 := I(w_1)$ and $I_2 := I(w_2)$ of distance $d(I_1, I_2) = \delta > 0$ and a number n, let w_0 be the shortest string such that $|w_0| \geqslant n$ and $d(I_1, I(w_0)) = d(I(w_0), I_2) > 2^{-n}$. Thus, the interval $I(w_0)$ has at most a length 2^{-n} and locates exactly in the middle between I_1 and I_2. This interval $I(w_0)$ is denoted by $M_n(I_1, I_2)$. The middle point of an interval I is denoted by $\mathrm{mid}(I)$.

Let (N_e^A) be a computable enumeration of all Turing machines with oracle A and suppose that N_e computes the computable functional Φ_e. By definition, a real x is Turing reducible to y if there exists an $i \in \mathbb{N}$ such that $x = \Phi_i^y$ i.e., $x(n) = \Phi_i^y(n)$ holds for all $n \in \mathbb{N}$. Thus, in order to construct two non-Turing equivalent reals x, y, we have to guarantee that $x \neq \Phi_i^y \vee y \neq \Phi_j^x$ for all i, j. To this end, we define a "length function" recording the maximal temporal agreement between (x, y) and (Φ_i^y, Φ_j^x) and try to destroy this agreement if it is possible to keep it finite. This is usually quite complicated if we consider all (i, j) in a priority construction. The following observation will simplify the matter a lot. If $x = \Phi_i^y$, then there is another computable functional Φ_j such that $x{\upharpoonright}n = \Phi_j^y(n)$ for all n, where $x{\upharpoonright}n$ is the initial segment of x of length n. (Here we identify a natural number n with the nth binary word under the length-lexicographical ordering.) Thus, two reals x and y are Turing equivalent iff there are $i, j \in \mathbb{N}$ such that x is (i, j)-*Turing equivalent* to y (denoted by $x \equiv_T^{(i,j)} y$) in the following sense:

$$(\forall n \in \mathbb{N}) \left(x{\upharpoonright}n = \Phi_i^y(n) \;\&\; y{\upharpoonright}n = \Phi_j^x(n) \right). \tag{1}$$

Although this is only a simple variation of usual Turing equivalence of the form $x = \Phi_i^y \;\&\; y = \Phi_j^x$, it connects the Turing equivalence to the topological structure of $\mathbb{R}$ more closely. More precisely, we have the following:

Lemma 2.1. *For any open interval $I_0 \subseteq \Sigma^\omega$, and any natural numbers i, j, t, there exist two open intervals $I \subseteq I_0$ and $J \subseteq \Sigma^\omega$ of the length at most 2^{-t} such that for any x and y with $x \equiv_T^{(i,j)} y$ we have*

$$x \in I \Longrightarrow y \in J \quad \text{and} \quad y \in J \Longrightarrow x \in I_0. \tag{2}$$

Proof. The idea of the proof is quite simple. If there does not exist $x \in I_0$ and $y \in \Sigma^\omega$ which are (i, j)-Turing equivalent, then we can choose $I = I_0$ and J arbitrarily. Otherwise, let $x \in I_0$ be a real number which is (i, j)-Turing equivalent to some $y \in \Sigma^\omega$. That is, $x{\upharpoonright}n = \Phi_i^y(n)$ and $y{\upharpoonright}n = \Phi_j^x(n)$ hold for all n. First, we choose a natural number $n \geqslant t$ such that $(x{\upharpoonright}n)\Sigma^\omega \subseteq I_0$. There exists $m > n$ such that the computation $\Phi_i^y(n)$ uses only the oracle information $y{\upharpoonright}m$. This means that, if $u \equiv_T^{(i,j)} v$ and $v \in I(y{\upharpoonright}m)$ then $u \in I(x{\upharpoonright}n) \subseteq I_0$. Thus, the interval $J := I(y{\upharpoonright}m)$ suffices for the second part of (2). Furthermore, choose a natural number $p > m$ large enough such that the computation $\Phi_j^x(m)$ queries only information on $y{\upharpoonright}p$. This guarantees that, $v \in I(y{\upharpoonright}p)$ if $u \equiv_T^{(i,j)} v$ and $u \in I(x{\upharpoonright}p)$. Thus, the intervals $I := I(x{\upharpoonright}p) \subseteq I(x{\upharpoonright}m) \subseteq I_0$ and $J := I(y{\upharpoonright}m)$ satisfy condition (2). Obviously, their lengths do not exceed 2^{-t}. □

Notice that, if the intervals I, J satisfy (the first part of) condition (2), then any real in I can only be (i, j)-Turing equivalent to reals in J. In other words, if the reals x, y satisfy $x \in I \;\&\; y \notin J$, then they are not (i, j)-Turing equivalent! Thus, to avoid the constructed real x being (i, j)-Turing equivalent to a given real y, it suffices to fix two interval pairs (I_l, J_l) and (I_r, J_r) according to the above lemma, then choose x from I_l whenever y does not seem in J_l and change x to I_r if y enters J_l and so on. This jump trick is called "escaping procedure". For convenience, we call an interval I (i, j)-*reducible* to J (denoted by $I \prec_{(i,j)} J$) if, for all $x, y, x \in I \wedge x \equiv_T^{(i,j)} y \Longrightarrow y \in J$. The second part of (2) guarantees that, if two distinct I-intervals are given, then the corresponding J-intervals are also distinct.

The proof of Lemma 2.1 is, of course, not constructive because we cannot effectively determine if there exists an $x \in I_0$ which is (i, j)-Turing equivalent to some y. However, given any $i, j, t \in \mathbb{N}$ and $w \in \Sigma^*$, we can always search for the intervals $I(u) \subseteq I(w)$ and $I(v)$ which satisfies condition (2). If we fail, we should guarantee that no real in $I(w)$ is (i, j)-Turing equivalent to any real. This leads to the following effective version of Lemma 2.1.

Lemma 2.2. *There exists a partial computable function* $\theta :\subseteq \mathbb{N}^3 \times \Sigma^* \to (\Sigma^*)^2$ *such that if there exist* $x \in I(w)$ *and* $y \in \Sigma^\omega$ *with* $x \equiv_T^{(i,j)} y$, *then*
(1) $\theta(i, j, t, w) \downarrow= (u, v)$;
(2) $l(I(u)), l(I(v)) \leqslant 2^{-t}$;
(3) $I(u) \prec_{(i,j)} I(v)$; *and*
(4) $(\forall x, y)(x \equiv_T^{(i,j)} y \;\&\; y \in I(v) \Longrightarrow x \in I(w))$.

Here, $\theta(i, j, n, w) \downarrow= (u, v)$ means that $\theta(i, j, n, w)$ is defined and has the value (u, v). Let M be a Turing machine which computes the function $\theta(i, j, n, w)$. By Lemma 2.2, for any interval $I_0 := I(w)$ and natural numbers i, j, t, we can compute $\theta(i, j, n, w)$ by running $M(i, j, n, w)$. If after s steps, the machine halts and outputs (u, v), then the interval $I := I(u)$ and $J := I(v)$ satisfy condition (2). Otherwise, if the machine never halts, then no real number of the interval I_0 is (i, j)-Turing equivalent to any real number.

3. Main result

In this section we prove the hierarchy theorem. For simplification we first prove a technical lemma which reformulates the condition of the main theorem.

Lemma 3.1. *Let* $\gamma > 1$ *be a real number and let* $f, g : \mathbb{N} \to \mathbb{N}$ *be monotonically increasing functions satisfying the following condition*:

$$(\forall c \in \mathbb{N})(\forall^\infty n)(f(\gamma n) + n + c < g(n)). \tag{3}$$

For any constant $a, b \in \mathbb{N}$, $\alpha < 1$ *and* $\beta > 1$ *such that* $\gamma > \beta/\alpha$, *the following holds*:

$$(\forall c \in \mathbb{N})(\forall^\infty n)(f(b + \beta n) + n + c < g(a + \alpha n)), \tag{4}$$

where "$\forall^\infty n$" *means* "*for almost all* n".

Proof. Suppose that $\gamma > 1$ be a constant and f, g be increasing functions which satisfy condition (3). Let $a, b \in \mathbb{N}$, $\alpha < 1$ and $\beta > 1$ be constants such that $\gamma > \beta/\alpha$. By the density of real numbers, there is an α_1 such that $\alpha > \alpha_1 > 1$ and $\gamma > \beta/\alpha_1 > \beta/\alpha$. Then, for a sufficiently large n we have

$$\begin{aligned} f(b + \beta n) + n + c &= f(\beta/\alpha_1(b\,\alpha_1/\beta + \alpha_1 n)) + n + c \\ &\leqslant f(\gamma(b\,\alpha_1/\beta + \alpha_1 n)) + (b\,\alpha_1/\beta + \alpha_1 n) + c \\ &\leqslant g(b\,\alpha_1/\beta + \alpha_1 n) \\ &\leqslant g(a + \alpha n). \qquad \square \end{aligned}$$

Now we prove our main result as follows.

Theorem 3.2. *Let $f, g : \mathbb{N} \to \mathbb{N}$ be two monotonically increasing computable functions such that $g(n+1) \geqslant g(n) + 2$ and*

$$(\forall c \in \mathbb{N})(\forall^{\infty} n)(f(\gamma n) + n + c < g(n)) \tag{5}$$

for some constant $\gamma > 1$. Then there exists a g-bc real x which is not Turing equivalent to any f-bc real.

Proof. Given two monotonically increasing computable functions $f, g : \mathbb{N} \to \mathbb{N}$ which satisfy condition (5) for a constant $\gamma > 1$, choose two rational numbers $\frac{1}{2} < \alpha < 1$ and $\beta > 1$ such that $\gamma > \beta/\alpha$. By Lemma 3.1, for any constants $a, b, c \in \mathbb{N}$, we have $f(b + \beta n) + n + c < g(a + \alpha n)$ for almost all n.

We construct a computable sequence (x_s) of rational numbers converging g-bounded effectively to a real number x which is not Turing equivalent to any f-bc real number. That is, for any computable sequence (z_s) of rational numbers, if it converges f-bounded effectively, then its limit is not (i, j)-Turing equivalent to x for any pair (i, j) of natural numbers. In other words, x satisfies all of the following requirements:

$$R_{\langle i,j,k \rangle} : \text{if} (\varphi_k(s))_s \text{ converges } f\text{-bounded effectively to } y_k, \text{ then } x \not\equiv_T^{(i,j)} y_k$$

for $i, j, k \in \mathbb{N}$, where $(\varphi_k(s))_s$ is a computable enumeration of all partial computable functions $\varphi_k :\subseteq \mathbb{N} \to \mathbb{D}$.

To satisfy a single requirement R_e for $e = \langle i, j, k \rangle$ we fix an interval I_{e-1} as the base interval of R_e and try to find a subinterval $I_e \subset I_{e-1}$ such that any $x \in I_e$ satisfies the requirement R_e. Such an interval I_e is called a *witness interval* of R_e. Thus, our goal is to find a correct witness interval for R_e.

As a default candidate we consider first the interval $L(I_{e-1})$, the second subinterval of the partition of I_{e-1} into eight equidistant parts. If no element of this interval is (i, j)-Turing equivalent to some real number y, then we are done. Otherwise, by Lemma 2.2, we can effectively find a pair (I_l, J_l) of intervals such that $I_l \subseteq L(I_{e-1})$ and $I_l \prec_{(i,j)} J_l$. If the sequence $(\varphi_k(s))_s$ does not enter the interval J_l, then the interval I_l is a correct witness interval of R_e and we are done. Suppose that the sequence $(\varphi_k(s))_s$ does enter the interval J_l. Then we consider the interval $R(I_{e-1})$ as a new candidate of the witness interval of R_e. Analogously, either $R(I_{e-1})$ is a correct witness interval of R_e or we can find another pair (I_r, J_r) of intervals such that $I_r \subseteq R(I_{e-1})$ and $I_r \prec_{(i,j)} J_r$. In this case, if the sequence $(\varphi_k(s))_s$ enters the interval J_r, then we can go back to the old interval pair (I_l, J_l) to escape the sequence $(\varphi_k(s))_s$ and so on. This escaping technique works well if the intervals J_r and J_l have a positive distance of at least, say, 2^{-n}, because at most $f(n)$ jumps are needed to find a correct witness interval of R_e from I_l or I_r if the sequence $(\varphi_k(s))_s$ converges f-bounded effectively.

Unfortunately, we cannot guarantee that $d(J_l, J_r) \neq 0$ so far and if $d(J_l, J_r) = 0$, then the sequence $(\varphi_k(s))$ can enter the intervals J_r and J_l alternatively infinitely often. In this case the escaping strategy described above fails. To solve this problem, we consider a third pair (I_m, J_m) of intervals as follows. Since the intervals I_l and I_r do have a positive distance, we can choose a new interval I'_m between I_l and I_r. Again, either the interval I'_m is already a correct witness interval of R_e (if no element of I'_m is (i, j)-Turing equivalent to any real number), or we arrive at a pair (I_m, J_m) of intervals such that $I_m \subset I'_m$ and $I_m \prec_{(i,j)} J_m$. Because all three intervals I_l, I_m and I_r are disjoint, the intervals J_l, J_m and J_r are also disjoint by condition 4 of Lemma 2.2. From three disjoint intervals J_l, J_m and J_r, we can find two, say J_l, J_r, of positive distance. By means of the interval pairs (I_l, J_l) and (I_r, J_r) we can find a correct witness interval for R_e by the above escaping technique. If we denote the candidate of witness interval of R_e at stage s by $I_{e,s}$ and define x_s as the middle point of $I_{e,s}$, then the sequence (x_s) converges and its limit x satisfies obviously the requirement R_e.

However, the computable sequence (x_s) constructed in this way does not necessarily converge g-bounded effectively. Suppose that the minimal distance between the intervals J_l and J_r is bounded below by 2^{-b} for some $b \in \mathbb{N}$, i.e., $2^{-b} \leqslant d(J_l, J_r)$. Then the number of necessary jumps in the above strategy is bounded by $f(b)$, if the sequence $(\varphi_k(s))_s$ converges f-bounded effectively. On the other hand, the sizes of the jumps contributed by this strategy are bounded by $D(I_l, I_r)$. Let a be the maximal natural number such that $2^{-a} \geqslant D(I_l, I_r)$. In general, there is no relationship between the numbers a and b available. The following strategy of "interval distance reduction" will introduce a sufficient relation between the two kinds of interval distances, so that the number of jumps between the I-intervals is bounded by the allowed number of jumps of (φ_k) between the J-intervals.

To explain the strategy of "interval distance reduction", suppose that we have two interval pairs (I_l, J_l) and (I_r, J_r) such that $D(I_l, I_r) \leqslant 2^{-a}$ and $d(J_l, Jr) \geqslant 2^{-b}$. Let $I_l = (u_1, v_1)$, $I_r = (u_2, v_2)$, $J_l = (u_3, v_3)$ and $J_r = (u_4, v_4)$. Assume that the intervals I_l and J_l are located on the left side of the intervals I_r and J_r (on the real axis), respectively, and assume further that the lengths of the intervals I_l and I_r are less than $D(I_l, I_e)/4$. (If it is not the case, we can choose new subinterval pairs according to Lemma 2.2 with this property.) Since $\alpha < 1$ and $\beta > 1$, we can find a natural number $k \geqslant 2$ such that

$$2^{-1} - 2^{-k} \geqslant 2^{-\beta} \quad \text{and} \quad 2^{-1} + 2^{-k} \leqslant 2^{-\alpha}. \tag{6}$$

Let I be a rational interval of length less than $2^{-k} D(I_l, I_r)$ located in the middle of the interval (u_1, v_2). Then, either I is a correct witness interval of R_e or we can find an interval pair (I_m, J_m) such that $I_m \subseteq I$, $l(J_m) \leqslant 2^{-k} d(J_l, J_r)$ and $I_m \prec_{(i,j)} J_m$. Of course, the interval J_m is not necessarily located between J_l and J_r. But we have always the following

$$\max\{d(J_l, J_m), d(J_l, J_m)\} \geqslant (2^{-1} - 2^{-k}) d(J_l, J_r) \geqslant 2^{-(b+\beta)}$$

and

$$\max\{D(I_l, I_m), D(I_l, I_m)\} \leqslant (2^{-1} + 2^{-k}) D(I_l, I_r) \leqslant 2^{-(a+\alpha)}.$$

Suppose w.l.o.g. that $d(J_l, J_m) \geqslant d(J_r, J_m)$ and denote the interval pairs (I_l, J_l) and (I_m, J_m) by (I_l^1, J_l^1) and (I_r^1, J_r^1), respectively. Then we have

$$D(I_l^1, I_r^1) \leqslant 2^{-(a+\alpha)} \;\&\; d(J_l^1, J_r^1) \geqslant 2^{-(b+\beta)}. \tag{7}$$

This finishes the first step of the "interval distance reduction". If we repeat this reduction n times, we will either stop in some $t \leqslant n$ steps at a correct witness interval of R_e or arrive at interval pairs (I_l^n, J_l^n) and (I_r^n, J_r^n) with the following properties

$$I_l^n \prec_{(i,j)} J_l^n \;\&\; I_r^n \prec_{(i,j)} J_r^n \tag{8}$$

and

$$D(I_l^n, I_r^n) \leqslant 2^{-(a+n\alpha)} \;\&\; d(J_l^n, J_r^n) \geqslant 2^{-(b+n\beta)}. \tag{9}$$

If the number n is chosen carefully such that $f(b + \beta n) + n + c < g(b + \alpha n)$ holds, where c denotes the current stage s of the construction, i.e., we have constructed the finite sequence

$(x_t)_{t \leqslant s}$ which has, of course, at most $c := s$ jumps of any size, then we can apply the standard escaping technique to find a correct witness interval I_l^n or I_r^n in at most $f(b + n\beta)$ further steps. In this way, we can ensure that the sequence (x_s) converges g-bounded effectively.

To satisfy all requirements R_e simultaneously we need a finite injury priority construction. We say that a requirement R_i has a higher priority than R_j if $i < j$. At the beginning of the construction let $I_{-1} := (0; 1)$ be the base interval for the requirement R_0 and we try to find a witness interval $I_0 \subset I_{-1}$ by the strategy described above. Our first candidate is the interval $I_0^0 := L(I_{-1})$. If this is a correct witness interval for R_0, then we can set $I_0 := I_0^0$ and search further for the witness interval $I_1 \subset I_0$, and so on. However, I_0^0 may be a wrong candidate and in this case we choose a new candidate $I_0^1 \subset I_0^0$ and an interval J_0^1 according to Lemma 2.2 such that $I_0^1 \prec_{(i_0, j_0)} J_0^1$, where $\langle i_0, j_0, k_0 \rangle = 0$. The interval I_0^1 may be a wrong candidate too if the sequence $(\varphi_{k_0}(s))$ enters the interval J_0^1. In this case, we consider $I_0^2 := R(I_{-1})$ and eventually $I_0^3 \subset I_0^2$ and so on. Thus, the candidate for I_0 can be changed many times. Although we arrive at a correct witness interval I_0 in finitely many steps as mentioned above, we cannot confirm in finitely many steps which candidate interval is a correct one. In other words, to tell certainly whether a candidate interval is a correct witness interval or not, we need essentially infinitely many steps. This means that, if we hang on finding a correct witness interval of R_0, then we have no chance to deal with other requirements any more. Therefore, we have to begin our search for a witness interval I_1 of R_1 before we have definitely a correct witness interval for R_0. For example, if I_0^t is the current candidate of R_0 and no evidence shows that it is not correct, then we can carry on with our strategy for R_1 on the interval I_0^t. If, however, at a later stage we find that I_0^t is not a proper candidate for R_0, then we change the base interval for R_1 to I_0^{t+1} and the strategy for R_1 has to be restarted on this interval newly. What we have done for R_1 on I_0^t is destroyed. (We say that R_1 is injured.) Since R_0 changes its candidate intervals only finitely often, we do have a chance to finish a strategy for R_1 eventually. Analogously, all requirements can be satisfied in this way.

In general, denote by $I_{e,s}$ the candidate interval for R_e at stage s in the construction. Then, at any stage s, we have a maximal finite sequence $I_{0,s} \supset I_{1,s} \supset \cdots \supset I_{k_s,s}$ of intervals consisting of all the current valid candidate intervals for $R_0, R_1, \ldots, R_{k_s}$, respectively. For any e, the limit $I_e := \lim_{s \to \infty} I_{e,s}$ exists and it is a correct witness interval for R_e such that $I_e \subset I_{e-1}$. Let x_s be the middle point of the interval $I_{k_s,s}$. Then the computable sequence (x_s) of rational numbers converges to x which locates in all witness intervals of R_e and hence it satisfies all R_e. The only problem left now is, whether the sequence (x_s) converges g-bounded effectively?

Let us look at the strategy for a single requirement R_e on the base interval I_{e-1} of a length less than $2^{-n_{e-1}}$ and calculate first how many jumps can be contributed by this strategy. Before the interval distance reduction only c_1 jumps are needed for a constant $c_1 \leqslant 8$. Suppose now that, we have two interval pairs (I_l, J_l) and (I_r, J_r) with $D(I_l, I_r) \leqslant 2^{-a}$ and $d(J_l, J_r) \geqslant 2^{-b}$. Then choose an n'_e large enough such that $f(b + \beta n'_e) + n'_e + c_1 < g(a + \alpha n'_e)$ and begin the interval distance reduction procedure. At most n'_e jumps can be caused by this procedure. It is followed immediately by the escaping procedure which can cause at most $f(b + \beta n'_e)$ jumps. In summary, at most $f(b+\beta n'_e)+n'_e+c_1 \leqslant g(a+\alpha n'_e)$ jumps can occur. Let $n_e := a+\alpha n'_e$, then the strategy for R_e can contribute at most $g(n_e)$ jumps of size larger than 2^{-n_e}. Suppose that n_{e-1} satisfies $g(n_{e-1}) \geqslant c_1$. Then for any $k \leqslant n_e - n_{e-1}$, the interval distance reduction causes at most $2k$ jumps of sizes larger than $2^{-(n_{e-1}+k)}$ since $\alpha > \frac{1}{2}$. This implies that, if $n_{e-1} \leqslant n < n_e$, then the total number of jumps larger than 2^{-n} is bounded by $c_1 + 2(n - n_{e-1}) \leqslant g(n_{e-1}) + 2(n - n_{e-1}) \leqslant g(n_{e-1} + (n - n_{e-1})) = g(n)$. This shows that, our strategy succeeds for a single requirement.

To consider all requirements simultaneously, we should avoid that the jumps of similar lengths are used by actions for different requirements. For example, at stage s, we arrive at the interval

$I_{e-1,s}$ and start the action for R_e by defining a new interval $I_{e,s+1} \subset I_{e-1,s}$. We can choose the interval $I_{e,s+1}$ in such a way that its length is less that 2^{-k} and all jumps of the finite sequence $(x_v)_{v \leqslant s}$ constructed sofar are larger than 2^{-k}. This guarantees that all jumps constructed later in the interval $I_{e,s+1}$ do not share the length of the jumps constructed before the stage s. Here, two jumps share a length means that both of their lengths locate in the interval $[2^{-m}; 2^{-m+1})$ for some $m \in \mathbb{N}$. The problem is, the jumps in the interval $I_{e,s+1}$ can share the length with jumps constructed later by the interval distance reduction or escaping procedure for R_i $(i \leqslant e)$. For the case of interval distance reduction, it can happen at most two times for each $i \leqslant e$ because $\alpha > \frac{1}{2}$ and can be bounded already in the definition of $I_{e,s+1}$. Sharing length with the jumps in the escaping procedure should be strictly forbidden by choosing a proper number m_e of steps of interval distance reductions. That is, besides the condition $f(b + \beta m_e) + m_e + c < g(b + \alpha m_e)$, $2^{-(b+\alpha m_e)}$ should be smaller than all jumps constructed so far. Whenever m_e is defined, any jumps constructed later on for any R_i with $i > e$ are not allowed to share this length. This guarantees that the sequence (x_s) indeed converges g-bounded effectively.

The details of the formal construction are omitted here. □

References

[1] K. Ambos-Spies, K. Weihrauch, X. Zheng, Weakly computable real numbers, J. Complexity 16 (4) (2000) 676–690.

[2] R.G. Downey, Some computability-theoretic aspects of reals and randomness, in: The Notre Dame Lectures, Lecture Notes in Logistics, vol. 18, Assoc. Symbol. Logic, Urbana, IL, 2005, pp. 97–147.

[3] R. Downey, G. Wu, X. Zheng, Degrees of d.c.e. reals, Math. Logic Quarterly 50 (4/5) (2004) 345–350.

[4] Y.L. Ershov, A certain hierarchy of sets. i, ii, iii. (Russian), Algebra Logika 7(1) (1968) 47–73; 7(4) (1968) 15–47; 9 (1970) 34–51.

[5] R. Rettinger, X. Zheng, On the Turing degrees of the divergence bounded computable reals, in: CiE 2005, June 8–15, Amsterdam, The Netherlands, Lecture Notes in Computer Science, vol. 3526, Springer, Berlin, 2005, pp. 418–428.

[6] R.M. Robinson, Review of "Peter R. Rekursive Funktionen", J. Symbolic Logic 16 (1951) 280–282.

[7] R.I. Soare, Cohesive sets and recursively enumerable Dedekind cuts, Pacific J. Math. 31 (1969) 215–231.

[8] A.M. Turing, On computable numbers with an application to the "Entscheidungsproblem", Proc. London Math. Soc. 42 (2) (1936) 230–265.

[9] X. Zheng, Recursive approximability of real numbers, Math. Logic Quart. 48 (Suppl. 1) (2002) 131–156.

[10] X. Zheng, On the divergence bounded computable real numbers, in: T. Warnow, B. Zhu (Eds.), Computing and Combinatorics, Lecture Notes in Computer Science, vol. 2697, Springer, Berlin, 2003, pp. 102–111, COCOON 2003, July 25–28, 2003, Big Sky, MT, USA.

[11] X. Zheng, On the Turing degrees of weakly computable real numbers, J. Logic Comput. 13 (2) (2003) 159–172.

[12] X. Zheng, R. Rettinger, Weak computability and representation of reals, Math. Logic Quart. 50 (4/5) (2004) 431–442.